

A Supplementary materials

We provide detailed prompts utilized throughout the PANGAEA framework. We first present the step-by-step prompts used in our initial motivation study (§ 3.1). Subsequently, we provide the structured prompts employed within our three-stage framework, leveraging the Prompt Writer (§ 3.2). Finally, we include the annotation prompts for each benchmark dataset evaluated in our experiments.

A.1 Step-by-step prompts for motivation study (§ 3.1)

Our initial proof-of-concept (Motivation) prompts were structured as follows. In this stage, data was generated in an end-to-end manner, where d_{g_i} represents **the irrelevant general data** instance and d_{s_i} denotes **the domain-specific source data** instance.

GSM8K prompt

You are an expert at transforming general questions into domain-specific, math-related questions. Your task is to generate only the transformed math question without including any answers or solutions.

```
### STEP 1: Analyze and Understand the General Question
- Fully understand the general question's context, key concepts, and
quantitative elements.
- Identify core topics (e.g., measurement, comparison, probability) and
specific details that inspire mathematical transformations.
```

```
### STEP 2: Refer to Domain-Specific Question (for Inspiration Only)
- Use the domain-specific question to understand common formulations in
mathematical contexts.
- Extract core mathematical concepts and focus on transformation, not
duplication.
- Frame questions encouraging mathematical reasoning (e.g., multi-step
calculations, logic, real-world applications).
```

```
### STEP 3: Generate the Transformed Math Question
- Create an original math question retaining general question context.
- Integrate scenario, characters, or objects clearly.
- Introduce mathematical challenges explicitly requiring calculations,
comparisons, or probabilistic/logical reasoning.
- Avoid solutions or numerical answers entirely.
```

Response Rule: Generate only the Transformed Domain Question without any answer, explanation, or solution.

Output Format:

Transformed Domain Question: [Write your transformed question here.]

Task:

- General Question:

```
"""
{dgi}
"""
```

- Domain Question: """

```
{dsi}
"""
```

Figure 7: **Motivation GSM8K prompt.** Prompt used for GSM8K Benchmark Generation.

```

You are an expert in transforming General Data and existing MedQA-style
clinical questions into high-quality, diverse, and complex MedQA-style clinical
questions. Generate realistic USMLE-style multiple-choice questions.
---
### Step 1: General Data Analysis & Clinical Scenario Transformation
- Convert General Data into clinical elements (demographics, lab values, timelines,
events, exposures, physiological processes).
- Transform key events into realistic clinical scenarios (accidents, exposures,
temporal shifts).
- Reinterpret events into clinical contexts (symptoms, history, diagnostics).
- Ensure clinical relevance (emergency medicine, infectious disease, cardiology).
### Step 2: Referencing Existing Domain Question Structure
- Refer to existing MedQA questions (structure, relevance, complexity).
- Recreate and diversify scenario, symptom progression, diagnoses.
- Introduce new events, mechanisms, injuries, exposures, patient history.
### Step 3: Generate Transformed Data (Scenario + MCQs)
1. Develop Complex Clinical Scenario
- Merge General Data with MedQA style for realistic patient scenarios
(age, gender, travel, medication, family history, exposures).
- Add disease progression, comorbidities, side effects, diagnostic errors.
2. Create New Multiple-Choice Questions
- Integrate General Data and MedQA clinical elements for unique scenario.
- Provide 4 answer options (1 correct, 3 relevant distractors).
Step 4: Final Output (Restrictions)
- Ensure accuracy, originality, logical consistency.
- Strictly output New Clinical Question and 4 options only.
- No explanations, comments, answers, reasoning.
### Output Format:
New Clinical Question:
Question: <Clinical question here>
Choose one of the following:
A. <Option 1>
B. <Option 2>
C. <Option 3>
D. <Option 4>
---
### Key Considerations:
- Convert General Data into medical data (timelines, labs, exposures).
- Reference existing MedQA questions but create original scenario.
- Diversify question types (diagnosis, treatment, prognosis, mechanisms).
- Ensure distractors are realistic and clinically relevant.
---
Task
General Question:
"""
{dgi}
"""
existing MedQA-style clinical question:
"""
{dsi}
"""

```

Figure 8: **Motivation MedQA prompt.** Prompt used for MedQA Benchmark Generation.

You are an expert in transforming General Data (general knowledge, everyday questions, or non-financial text) into structured Financial Data and generating high-quality, diverse, and complex financial reasoning problems.

--

Your Enhanced Approach Must Include:

Step 1. General Data → Financial Context Mapping

- Convert General Data into a finance-oriented scenario.
- Identify relevant financial events, business implications, investment scenarios, or economic contexts that relate to the General Data.
- Ensure financial scenario is logically consistent, realistic, and distinct from Existing FinQA-style financial reasoning question.

Step 2. Financial Context → Structured Financial Data Generation

- Create realistic financial data (revenue, costs, tax, investments, stocks).
- Include at least two different financial variables for complexity.
- Generate data solely from financial context derived from General Data.

Step 3. Financial Data → Unique FinQA-style Question Generation

- Formulate multi-step calculations, financial analysis, market evaluation, investment decision-making, or risk assessment question.
- Ensure complete originality and no copying from existing FinQA questions.
- Use new business scenarios derived from General Data.

Output Restriction:

- Strictly generate New Financial Context and Question only.
- Maintain similar length and structure as Existing FinQA-style question.
- Do NOT reference existing company names or financial details.
- Ensure financial scenario derived entirely from General Data.
- No explanations, comments, or notes.

--

Output Format:

New Financial Context and Question:

Please answer the given financial question based on the context.

Context: <Generated financial context with relevant numerical table data>

Question: <Generated financial reasoning question>

Enhanced Data Processing Flow:

- Extract key themes from General Data.
- Map themes to financial events (corporate, market, investments).
- Generate structured Financial Data (revenue, costs, indicators).
- Create unique financial reasoning problem (complexity, originality).

--

Strict Constraints:

- DO NOT copy company names or figures from Existing FinQA question.
- All data derived solely from General Data.
- Ensure complete originality.
- Diversify numerical relationships and economic insights.

Task

General Data:

"""

$\{d_{gi}\}$

"""

Existing FinQA-style financial reasoning question:

"""

$\{d_{si}\}$

"""

Figure 9: Motivation FinQA prompt. Prompt used for FinQA Benchmark Generation.

Step-by-Step Prompt Example for Custom DSL:
 You are an expert in converting General Data and a Reference String-Manipulation Task into a fresh benchmark item that matches the reference style while remaining fully original.

```

---
## Inputs:
General Data:
"""
{dgi}
"""
Reference String Task:
"""
{dsi}
"""
---
### STEP 1 · General Data Analysis & Building-Block Construction *(keep private)*
1. Invent the initial string S -- 5-50 chars, quoted, contains  $\geq 1$  separator.
2. Write one Goal sentence describing the intended transformation of S.
3. Select 3-6 operations (split, join, reverse, str(n), reverse_tokens, replace, substr, upper, lower, append, prepend).
4. Draft the op chain (op(arg) in order).
5. Compute the target string T by mentally executing the chain.
Keep S, Goal, Ops, Chain, and T hidden; never expose them.
---
### STEP 2 · Referencing Existing Task Structure
- Examine Reference Task for leading verb/phrase and sentence length & tone.
- Craft one instruction sentence  $\leq 60$  words, starting with that leading verb/phrase, quotes S exactly once, uses vivid verbs, no ops or intermediate math.
--
### STEP 3 · Projection & Final Benchmark Generation
Goal: Publish final benchmark item in strict two-line format.
Header: New String-Transformation Task:
Problem-statement sentence (verbatim from Step 2).
Desired Output: "<T>" (directly after problem sentence).
No Extras: Exactly two lines shown, no comments or explanations.
---
### HARD RULES
Reveal nothing from Step 1. Exactly three printed lines (header + two content lines). Task requires 3-6 implicit operations. Never copy wording or digits from the reference; mirror only opener and sentence length.
```

Figure 10: **Motivation CDSL prompt.** Prompt used for CDSL Benchmark Generation.

837 **A.2 PANGAEA: Profiling prompt (§ 3.2)**

838 The prompt used in the initial profiling stage of our three-stage PANGAEA framework (§ 3.2) is
839 presented below. In this stage, we provided our scarce domain-specific source data (\mathcal{D}_s) to OpenAI's
840 o1 model and requested a detailed analysis of the dataset. Specifically, the model was instructed to
841 identify key characteristics, structural patterns, and thematic elements of the provided data.

Iteration 1: Initial analysis

"<Insert domain-specific source dataset \mathcal{D}_s here>"

User:

You are an analyst tasked with transforming irrelevant general-domain data into {domain_name} - specific data. Carefully analyze the 100 provided domain-specific source examples. Identify their key characteristics, structural patterns, and thematic elements. Suggest what types of information should be extracted or transformed from unrelated general datasets to create useful domain_name data.

o1:

<model's initial analysis and suggestions>

Iteration 2: Refinement based on generated data

User:

Following your suggested approach, we generated this synthetic example:

"<first synthetic data example>"

Given this outcome, what additional information or refinements do you suggest extracting or transforming from general data to better align with the domain_name domain?

o1:

<model's refined analysis and further suggestions>

⋮

(The iterative process continues until desired data quality is achieved.)

Figure 11: **Synth-profiling prompt.** Example dialogue illustrating the iterative profiling process used in the PANGAEA framework.

842 An illustrative example of the profiling prompt dialogue is presented in Figure 11. Based on the
843 initial analysis, the model generated structured guidelines describing how irrelevant general-domain
844 data could be effectively transformed into high-quality, domain-specific synthetic data. This profil-
845 ing process was iterative: synthetic examples were first generated following the guidelines provided
846 by the model, their quality was evaluated, and instructions were progressively refined based on feed-
847 back if the generated data did not sufficiently meet the desired criteria. This iterative, conversational
848 approach ensured continuous improvement of instructions and consistent generation of high-quality
849 synthetic data.

850 A.3 PANGEA: Prompt writer prompts (§ 3.2)

851 The prompt used in the second stage of the PANGEA framework (§ 3.2), which employs the Prompt
 852 Writer to generate structured Synth-Guide Blocks (τ), is presented below. Only irrelevant general
 853 data is provided as input to the Prompt Writer.

854

GSM8K Prompt-Writer's prompt

```

You are an expert at distilling everyday descriptions into concise,
GSM8K-ready
quantitative building blocks. Your output feeds next stage, converting
these
blocks into a 2-4 sentence word problem solvable by a 5th- to 8th-grader
in
multiple arithmetic or logical steps.
### Inputs
General Question (raw text)
"""
{dgi}
"""
-----
### Choose a Kid-Friendly Scenario
- Restate one everyday clause students can picture
  (shopping, chores, snacks, simple travel).
- Remove unnecessary details.
### Select 3-5 Meaningful Numbers
- Prefer numbers in General Question.
- Invent realistic values if needed, supporting clear multi-step
  solution.
- Use everyday units (dollars, minutes, km, items, °C, simple percents).
### Define Symbols & Units
- Assign symbols A, B, C, (D, E) with brief label and explicit unit.
### Write 4-5 Simple Equations
- Use only +, -, ×, ÷, %, or single-step unit conversions.
- No new numbers; use chosen symbols or intermediate results.
### State the Target
- Specify final symbol to solve and its numeric meaning.
### Outline Step-by-Step Plan
- Provide one bullet per equation, in order (4-5 steps).
-----
### Output Format (use EXACTLY this template)
Quantitative Building Blocks:
- Scenario - <one kid-friendly clause>
- Symbols - <A = value unit - label>, <B = ...>, <C = ...>[, <D = ...>[,
  <E = ...>]]
- Equations - (1) ... = ...
  (2) ... = ...
  (3) ... = ...
  (4) ... = ...
  [(5) ... = ...]
- Target - <T = description of numeric goal>
- Plan - (1) ... → (2) ... → (3) ... → (4) ... [→ (5) ...]
### Hard Restrictions & Key Considerations
- Scenario familiar, everyday, age-appropriate; no fantasy/sci-fi.
- All numbers appear in Scenario and Equations.
- Exactly 4 or 5 equations, requiring at least four steps.
- Do NOT write/solve full word problem.
- No examples, explanations, or commentary.
- Begin exactly with "Quantitative Building Blocks:" and follow template
  strictly.

```

Figure 12: **Prompt Writer prompt for GSM8K.** This prompt instructs the model to analyze irrelevant general data and explicitly extract structured information relevant to arithmetic reasoning, including scenarios, equations, symbols, and solution plans for GSM8K.

855

856 MedQA Prompt-Writer's prompt

You are an expert at transforming General Data into clinically useful Clinical building blocks.

```
## General-to-Clinical Mapping
General Data:
"""
{dgi}
"""
---
### 1. Analyze General Data and extract every relevant
- numbers & ranges
- logical/causal relationships
- temporal progressions
- physical principles, legal scenarios, or major events.
### 2. Convert these elements into clinical building blocks, such as
- patient demographics, timelines, physiologic/metabolic processes
- lab values, vital-sign ranges, exposures, accidents, disease
progressions.
### 3. Output ONLY the transformed clinical building blocks in the exact
format below:
---
### Output Format
Clinical Building Blocks:
- Demographics - ...
- Timeline - ...
- Key clinical events - ...
- Lab / imaging patterns - ...
- Pathophysiologic principles - ...

### Hard Restrictions
- Do NOT generate questions, answer options, or explanations.
- Output must begin exactly with "Clinical Building Blocks:"
followed by the bullet list above.
```

Figure 13: **Prompt Writer prompt for MedQA.** This prompt guides the model to analyze unrelated general data and extract structured clinical elements tailored specifically for MedQA.

857

FinQA Prompt-Writer's prompt

You are an expert in translating General Data (non-financial facts or everyday scenarios) into a realistic, detailed, logically consistent financial scenario.

```
#####
### Task:
1. Carefully analyze the provided General Data.
2. Clearly translate the General Data into a concise financial scenario, explicitly
highlighting:
- Specific and realistic financial implications or investment contexts.
- Potential corporate decisions, economic outcomes, or market
environments logically derived from the data.
3. Scenario must be distinct, detailed, clear enough for later use in Stage 2
(structured financial data and FinQA-style questions).
4. Do NOT produce numerical tables, numeric values, or financial questions now.
5. Avoid closely replicating illustrative examples or specific contexts from prior
instructions; generate entirely original scenario uniquely based on provided
General Data.
#####
### Output (follow EXACTLY):
Financial Scenario:
<Clearly describe your original financial scenario derived from the provided
General Data. Include explicit descriptions of financial events, investment
opportunities, economic contexts, or corporate implications that are realistic
and logically consistent.>
#####
### Provided General Data:
"""
{dgi}
"""
```

Figure 14: **Prompt Writer prompt for FinQA.** This prompt directs the model to carefully analyze irrelevant general-domain data and extract a financial scenario specifically for FinQA.

858

```

You are an expert at distilling everyday descriptions into concise
String-Flow Building Blocks. Your output will feed the next stage, which
converts these blocks into a complete String-Flow DSL "recipe"
(code) that the interpreter can run.
--
## Stage 1 - General Description → String-Flow Building Blocks
General Text (raw):
"""
{dgi}
"""
### 1. Define the Initial String S
- Single quoted string, 5-50 chars.
- Include at least one separator or symbols.
- Remove irrelevant details.
### 2. State the Desired End-State T (plain words)
- Describe in one sentence the intended final form of S
(e.g., "reverse the items and join them with hyphens").
### 3. Select 3-6 Essential Operations
- Choose from split · join · reverse_str · reverse_tokens · replace
(" " to delete) · substr · upper · lower · append · prepend.
- Briefly list each operation with arguments if needed
(delimiter, replacement text, start/length, etc.).
- set (loading S) always implied, omit from the list.
### 4. Draft an Operation Chain
- List operations in order: (1) ... → (2) ... → (3) ... (3-6 steps).
- Each node: operation(arg).
### 5. Give the Target String T (concrete)
- Write exact resulting string after all operations (Stage 2
correctness checking).
---
#### Output Format (use exactly this)
String-Flow Building Blocks:
- S - "<initial string>"
- Goal - <plain-language description of desired result>
- Ops - <op1(arg)>, <op2(arg)>, <op3(arg)>[, ...]
- Chain - (1) ... → (2) ... → (3) ... [→ (4) ... → (5) ... → (6) ...]
- T - "<target string>"
#### Hard Rules & Tips
- Ops and Chain must match exactly in order and content.
- No unnecessary operations or arguments.
- substr indices: 0-based [start, length].
- upper/lower convert ASCII A-Z/a-z only.
- Everyday, age-appropriate contexts; no complex jargon.
- Do not add examples, explanations, or commentary.
- Begin your answer with "String-Flow Building Blocks:"
exactly as above, nothing else.

```

Figure 15: Prompt Writer prompt for CDSL. This prompt explicitly instructs the model to analyze unrelated general-domain data and extract structured elements required by the CDSL benchmark.

861 A.4 PANGAEA: Projection prompts (§ 3.2)

862 Using the following prompts, we performed the projection described in § 3.2. Here, d_{s_i} denotes the
 863 domain-specific source data instance, and the Synth-Guide Block (τ) generated by the Prompt
 864 Writer is provided as block, facilitating the generation of the final synthetic data.

866 GSM8K projection prompt

867

You are an expert in crafting GSM8K-style math word problems that require clear, multi-step quantitative reasoning.

Quantitative Blocks → Transformed Math Question

```
Inputs
- Quantitative Building Blocks
"""
{block}
"""
- Reference GSM8K Question (for tone & length only)
"""
{dsi}
"""
---
### Adapt Structure & Inject Realism
- Mirror only tone, length, complexity of reference; never copy content.
- Length: exactly 2 to 4 crisp sentences.
- Seamlessly weave Scenario, Entities, Numbers, Required relationships from Building Blocks into natural story.
- Embed each numeric value exactly once; avoid repeating numbers.
- No intermediate calculations shown or mentioned.

### Compose the New Math Problem (Projection)
1. Preserve Context
- Keep original setting/characters (or logical variants).
2. Embed Every Given Number
- All numeric values from Blocks must appear and be essential.
3. Demand Clear Multi-Step Challenge
- Solver must implicitly perform at least five distinct steps.
4. Avoid Superfluous Data
- Do not invent extra numbers; recombine given numbers if needed.
5. Maintain Numerical Consistency
- Respect units; conversions must be inferable (e.g., min→h, cm→m).
---
### Final Output & Hard Restrictions
- Output ONLY the problem text--no answers, hints, explanations.
- Begin exactly with "Transformed Domain Question:".
- Do NOT reuse exact wording or numeric values from reference GSM8K.
- Problem must require at least five distinct steps to solve.
- Length: exactly 2 to 4 concise sentences.
- Never repeat numbers; never expose arithmetic/intermediate results.
---
### Output Template
Transformed Domain Question: <sentence 1> <sentence 2> <sentence 3
(optional)> <sentence 4 (optional and must be explicit question)>
```

Figure 16: **Projection prompt for GSM8K.** This prompt explicitly directs the model to project the Synth-Guide Block onto GSM8K-specific source data, ensuring the generated synthetic examples.

You are an expert in crafting complex, USMLE-style (NBME-format) multiple-choice medical & clinical questions.

```
## Inputs
- Clinical Building Blocks
"""
{block}
"""
- Existing MedQA-style clinical question (structure reference only)
"""
{dsi}
"""
---

## Reference & Diversify Existing Question Structure
- Use existing question for format, difficulty, diagnostic flow only;
do NOT copy storyline.
- Recreate/diversify clinical context, symptom progression, diagnoses.
- Add clinically relevant events (injury mechanisms, toxic exposures,
detailed history) from Clinical Building Blocks.

### Projection & MCQ Creation (USMLE Blueprint)
1. Develop Complex Clinical Scenario
- Concise patient vignette (age, sex → chief-complaint → HPI → PE/labs).
- Insert ≥2 essential numeric data (vitals, labs, imaging).
- Integrate USMLE clues (pathognomonic signs, imaging findings, signature lab patterns,
POD-complications, drug toxicity).
- Include dynamic elements (progression, comorbidities, side-effects).
- Exposure history details (≥2: intensity mg/m3, duration, frequency,
route, protective equipment).

2. Create New Multiple-Choice Question
- Stem: ask either (a) most likely diagnosis or (b) next best step
/ appropriate management (only one).
- Provide exactly 4 answer options (A-D):
- One correct answer (current guidelines).
- Three high-quality realistic distractors (differentials or plausible
suboptimal choices).
- All options USMLE-standard medical concepts/actions.

### Final Output & Hard Restrictions
- Verify clinical accuracy, originality, data coherence, consistency.
- Output Restriction: ONLY New Clinical Question & 4 answer options.
- NO explanations, rationales, answer keys, extra text.

### Output Format
New Clinical Question:
Question: <clinical question text>
Choose one of the following:
A. <Option 1>
B. <Option 2>
C. <Option 3>
D. <Option 4>
---

### Key Considerations (USMLE Focus--do not output)
- Embed guiding numbers (Na+ 128 mEq/L, TSH 12 μIU/mL, EF 35%).
- Diagnosis distractors: common look-alike diseases.
- Management distractors: reasonable suboptimal or contraindicated
therapies.
- Follow current USMLE/NBME consensus guidelines.
```

Figure 17: Projection prompt for MedQA. This prompt explicitly instructs the model to transform the Synth-Guide Block into synthetic examples aligned with MedQA.

```

You are an expert in creating original, realistic, and complex FinQA-style financial reasoning
problems based on provided inputs.
---
### Task:
Using these inputs:
- Financial Scenario (from General Data)
- Existing FinQA-style Question (REFERENCE ONLY)

Generate a completely new, high-quality financial context and multi-step reasoning question
explicitly projecting Stage 1 Financial Scenario into a financial reporting narrative matching
Existing FinQA-style Question's style, complexity, and detail.
---
### STRICT Requirements:
(1) Main Narrative Context:
- Emulate Existing Question's narrative complexity with lengthy, complex
sentences and detailed explanations.
- Detailed definitions/explanations of  $\geq 2$  financial terms or acronyms.
- Use formal, passive-voice financial-reporting language ("was recognized,"
"were recorded," "is anticipated to be realized").
- Include  $\geq 1$  explicit note-reference clause ("net of tax," "see Note X").
- Describe financial events qualitatively, without repeating numbers.
(2) Financial Table (Markdown Format):
- Precisely replicate Existing Question's table dimensions (rows, columns).
- Use identical unit descriptor ("$ in millions," "$ in billions").
- Original yet realistic financial labels/numerical data closely
matching Existing Question's magnitude and complexity.
- Explicit markdown formatting matching Existing Question.
(3) Follow-up Narrative (Mandatory):
- Exactly match Existing Question's length, complexity, style, depth.
- Provide detailed qualitative insights without repeating table numbers.
- Explicitly address profitability trends, cost control effectiveness,
strategic impacts, or risk management considerations.
(4) Advanced, Multi-step Financial Reasoning Question:
- Require  $\geq 2$  sophisticated calculation steps (ratios, multi-year growth,
weighted averages, financial adjustments).
- Exactly match Existing Question's complexity, brevity, and clarity.
---
### STRICTLY AVOID:
- Copying exact wording, company names, numbers from Existing Question.
- Overly simplistic or ambiguous single-step questions.
- Repeating illustrative examples; all content must be original.
- Explanations, solutions, answers, additional commentary.
- Omitting or generalizing Follow-up Narrative.
---
### OUTPUT FORMAT:
New Financial Context and Question:
Context: <Explicitly structured narrative context exactly matching Existing FinQA Question's
detailed sentence complexity, narrative length, passive voice, financial explanations, note
references, and qualitative financial descriptions.>
<Markdown-formatted financial table precisely matching dimensions, complexity, numerical
magnitude, digit length, exact unit descriptor, with entirely original data from Stage 1
scenario.>
<Detailed Follow-up Narrative explicitly matching Existing Question's length, complexity,
style, analytical depth; provide financial insights without repeating table numbers.>
Question: <Explicitly stated advanced-level multi-step financial
reasoning question exactly matching Existing Question's brevity,
complexity, clarity, requiring  $\geq 2$  calculation steps.>
---
### Provided Inputs:
Financial Scenario:
"""
{block}
"""
Existing FinQA-style question (REFERENCE ONLY):
"""
{dsi}
"""

```

Figure 18: Projection prompt for FinQA. This prompt instructs the model to project the SynthGuide Block onto FinQA-specific source data to generate high-quality synthetic data for FinQA.

```

You are an expert at crafting concise, multi-step string-manipulation
tasks
for benchmark datasets.
### Inputs
- String-Flow Building Blocks
"""
{block}
"""
- Existing Generic String Task (style and format reference only)
"""
{dsi}
"""
---
### Use the Reference ONLY for Format & Tone
- Match length, phrasing style, and difficulty, but do NOT copy
storyline or wording.
- Recreate transformation using the Building Blocks.
+Style-Mirroring Rules
- Begin your sentence with the same leading verb or phrase
used in the reference task.
### Paraphrase Requirement
- Turn bland "Goal" line into vivid instruction (≤ 60 words).
- Mention initial string S exactly as given (including quotes,
separators, etc.).
- Do not reveal internal operation names.
- Prefer concrete verbs (strip, reorder, reverse, join).
---
### Create the New String-Transformation Task
1. Problem Statement
- One sentence satisfying Paraphrase Requirement.
2. Desired Output
- Present T in quotes on its own line, labeled
"Desired Output:".
### Output Restrictions
- Output only lines shown below--no explanations, extra text.
- Begin with exact header "New String-Transformation Task:".
- Preserve line breaks exactly.
---
### Output Format (copy exactly)
New String-Transformation Task:
<Problem Statement>

Desired Output: "<T>"

```

Figure 19: Projection prompt for CDSL. This prompt explicitly guides the model in projecting the Synth-Guide Block onto CDSL-specific source data to produce high-quality synthetic data for CDSL.

874 **A.5 Annotation prompts (§ 3.2)**

875 The annotation prompt is used to collect responses for the generated synthetic data. By specifying
876 a structured output format, this prompt facilitates both the model's chain-of-thought reasoning and
877 the parsing of responses, thereby simplifying performance evaluation.

878

879 **GSM8K annotation prompt**

You are an expert in mathematical problem solving. Your task is to provide a detailed and logically sound solution to the given math problem. Each response must be accurate and based on rigorous mathematical reasoning.

Response Rule

- Provide clear step-by-step solutions, including relevant mathematical principles and theorems.
- Explore multiple solution methods if applicable; compare their efficiency.
- Verify final answers through appropriate validation methods.
- Conclude clearly with "The final answer is: [Final Answer]" under the header ****Answer****.

Output Format

- ****Answer****: [Provide detailed mathematical explanation here, including step-by-step derivations, logical reasoning, and verification of calculations. Conclude explicitly with "The answer is: [Final Answer]".]

Your Task:

- ****Question****: {Question}

Figure 20: **Annotation prompt for GSM8K**. Prompt used to collect annotation responses for the GSM8K synthetic data.

880

881 **MedQA annotation prompt**

You are an expert in medical question answering. Your task is to provide a detailed, evidence-based response to the given multiple-choice medical question, accurate and aligned with up-to-date medical guidelines.

Response Rule

- Provide comprehensive explanation including relevant clinical reasoning.
- Analyze all answer choices, explaining correctness or incorrectness.
- Conclude clearly with final answer format:
"The answer is: [Answer Letter]. [Answer Option]" under the header ****Answer****.

Output Format

- ****Answer****: [Provide detailed medical explanation here, including clinical reasoning, differential diagnosis, and evidence-based references. Conclude explicitly with "The answer is: [Answer Letter]. [Answer Option]".]

Your Task:

- ****Question****: {Question}

Figure 21: **Annotation prompt for MedQA**. Prompt used to collect annotation responses for the MedQA synthetic data.

882

You are an expert in financial question answering. Provide detailed, evidence-based responses to given financial questions. Each response must be accurate, concise, and based on financial principles, accounting standards, and quantitative analysis.

Response Rule

- Provide step-by-step breakdown of calculations, relevant formulas, and financial reasoning.
- Clearly explain each step and how data points derive the final answer.
- Explicitly state and justify necessary assumptions.
- Conclude with Final Answer clearly boxed: `\boxed{}`.

Output Format

- **Answer**:

[Provide detailed financial explanation here, including step-by-step calculations, financial reasoning, and key insights.]

The answer is: `\[\boxed{[Final Answer]} \]`

Your Task:

- **Question**: {Question}

Figure 22: Annotation prompt for FinQA. Prompt used to collect annotation responses for the FinQA synthetic data.

You are an expert programming assistant familiar with a custom interpreter called `StringFlowInterpreter` (also known as `SousChef`). This interpreter uses cooking-themed commands to manipulate strings step-by-step.

Interpreter Command Quick Reference:

- `'pour' "string"`: Set main string (broth) to specified string.
- `'slice "delimiter"'`: Split broth into tokens (ingredients) by delimiter.
- `'stir "delimiter"'`: Join tokens (ingredients) into broth using delimiter.
- `'flip'`: Reverse broth string.
- `'toss'`: Reverse order of tokens (ingredients).
- `'season "old" "new"'`: Replace occurrences of substring old with new.
- `'fillet start length'`: Extract substring from broth (start, length).
- `'flambe'`: Convert broth to uppercase.
- `'simmer'`: Convert broth to lowercase.
- `'garnish "string"'`: Append specified string to broth.
- `'plate "string"'`: Prepend specified string to broth.
- `'taste_then "substring" label'`: If broth contains substring, jump to label.
- `'move_to label'`: Unconditionally jump to specified label.
- `'label LABEL_NAME'`: Define jump destination label.
- `'serve'`: Print current broth and end execution.

Given cooking-themed instruction, follow these steps:

1. Analyze and generate final command sequence according to `StringFlowInterpreter` rules.
2. Explain step-by-step in detail how final command sequence was derived.
3. Conclude with clearly indicated final generated command sequence:

output format:

<your step by step explanation>

The answer is:

""
<line by line final command sequence>
""

Important:

- Do NOT include original provided example; only step-by-step explanation clearly followed by final command sequence.
- In each command line, Do NOT include comments.

Your Task:

- `**Question:** {Question}`

Figure 23: **Annotation prompt for CDSL**. Prompt used to collect annotation responses for the CDSL synthetic data. An additional detailed task description was provided to ensure accurate annotation by the model due to the novel and domain-specific nature of CDSL tasks.

B Experimental details

B.1 Models and datasets

Models. The pre-trained weights used in our experiments are summarized below.

- **Llama3.2-1B:** <https://huggingface.co/meta-llama/Llama-3.2-1B>, licensed under Llama3.2 Community License.¹
- **Qwen2.5-1.5B:** <https://huggingface.co/Qwen/Qwen2.5-1.5B>, licensed under Apache 2.0 License.²
- **Gemma-2B:** <https://huggingface.co/google/gemma-2b>, licensed under Gemma License.³
- **Llama3.3-70B-Instruct:** <https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct>, licensed under Llama 3.3 Community License.⁴
- **Qwen2.5-72B-Instruct:** <https://huggingface.co/Qwen/Qwen2.5-72B-Instruct>, licensed under Apache 2.0 License.²
- **DeepSeek-V3-0324:** <https://huggingface.co/deepseek-ai/DeepSeek-V3-0324>, licensed under MIT License.⁵

Both our motivation study and the proposed three-stage PANGAEA framework employed the Llama3.3-70B-Instruct model for generating synthetic data. To ensure fair comparisons, we generated synthetic data for all baseline methods (Naive, Evol-Instruct, and DataTune) using the same Llama3.3-70B-Instruct model.

Annotations of the synthetic data were also performed using Llama-3.3-70B-Instruct, except for the CDSL. For CDSL, annotation accuracy with Llama3.3-70B-Instruct reached only 33.62%, despite detailed task instructions. Thus, we specifically employed the DeepSeek-V3 model to annotate synthetic data for the CDSL benchmark.

Due to budget constraints, subsequent experiments utilized smaller pre-trained base models: Llama3.2-1B, Qwen2.5-1.5B, and Gemma-2B. We conducted supervised fine-tuning (SFT) using the Alpaca template [30], training all models with full-parameter updates.

Datasets. The datasets used in our experiments are summarized below:

- **COT-Collection** [14] consists of 1.84 million Chain-of-Thought (CoT) augmented samples across 1,060 tasks derived from the Flan Collection, designed to induce CoT reasoning capabilities into language models.
- **GSM8K** [5] contains 8,792 arithmetic reasoning questions requiring multi-step logical reasoning and numerical calculations.
- **MedQA** [13] consists of over 12,000 standardized medical exam (USMLE) questions, assessing models’ medical knowledge and clinical decision-making abilities.
- **FinQA** [3] includes over 8,000 complex financial reasoning problems extracted from real-world financial reports, testing numerical analysis and financial comprehension.
- **CDSL** is introduced to evaluate models under severe data scarcity and extreme domain novelty. It involves transforming input strings into target outputs using a novel interpreter, `StringFlowInterpreter`, with entirely unseen commands and functions. The dataset contains only 100 manually curated training examples and 345 test instances.

Since our experimental setting involves an extreme scenario of data scarcity in highly specialized domains without any related general-domain data available, we constructed our irrelevant general dataset (\mathcal{D}_g) from the CoT-Collection. To ensure strict irrelevance, we manually removed samples related to mathematics, medical, finance, and coding tasks by using

¹https://www.llama.com/llama3_2/license/

²<https://www.apache.org/licenses/LICENSE-2.0>

³<https://ai.google.dev/gemma/terms>

⁴https://www.llama.com/llama3_3/license/

⁵<https://github.com/deepseek-ai/DeepSeek-V3-0324/blob/main/LICENSE>

931 Llama-3.3-70B-Instruct and Qwen/Qwen2.5-72B-Instruct across five different
932 random seeds.

933 For the GSM8K, MedQA, and FinQA benchmarks, we randomly selected 100 samples from each
934 dataset’s training split to form our domain-specific source dataset. In the case of the CDSL, which
935 includes 100 manually curated training examples and 343 test samples specifically constructed for
936 evaluating models in novel and specialized domain settings, we directly utilized the entire set of 100
937 training samples as our domain-specific source dataset (\mathcal{D}_s).

938 Our proposed benchmark, Custom Domain-Specific Language (CDSL), is specifically designed to
939 evaluate language models’ capabilities in highly specialized domains and extreme data scarcity set-
940 tings. The task involves transforming an input string into a desired output by generating executable
941 code conforming to a novel interpreter (`StringFlowInterpreter`, detailed in [Algorithm 2](#)).
942 Unlike conventional programming languages or commonly used scripting syntax, our interpreter
943 features uniquely crafted cooking-themed commands, enhancing domain specificity and novelty.

944 This cooking metaphor-based interpreter maintains internal state representations (such as `broth`
945 for strings and `ingredients` for tokens) and executes instructions through commands named
946 after culinary actions (e.g., `slice`, `stir`, `flambe`). This distinctive structure not only empha-
947 sizes domain novelty but also ensures the synthetic generation process must capture highly specific
948 instructions and contextually relevant transformations, effectively evaluating the adaptability of syn-
949 thetic data augmentation methodologies in novel environments.

950 Specifically, for annotating the CDSL benchmark, we employed the `Deepseek-V3` model. The
951 generated code, enclosed within triple backticks (`'''`), was extracted and compiled. Instances
952 where the code failed to compile or produced outputs inconsistent with the labels were filtered out.

Algorithm 2 CDSL: Interpreter

```
1 class StringFlowInterpreter:
2     def __init__(self):
3         self.broth = "" # str_val
4         self.ingredients = [] # tokens
5         self.stations = {} # labels
6         self.pc = 0
7         self.recipe = [] # program
8         self.cooking = True # running
9
10    def prep(self, recipe_str): # parse → prep
11        for line in recipe_str.strip().split('\n'):
12            dish = line.strip()
13            if not dish:
14                continue
15            if re.match(r'^[\w]+$', dish):
16                self.stations[dish[:-1]] = len(self.recipe)
17            else:
18                step, *args = self.dice(dish)
19                self.recipe.append((step, args))
20
21    def dice(self, line): # _split_line → dice
22        return re.findall(r'\"[^\"]*\|\"|S+', line)
23
24    def cook(self): # run → cook
25        self.pc = 0
26        while self.cooking and self.pc < len(self.recipe):
27            step, args = self.recipe[self.pc]
28            self.saute(step, args) # execute → saute
29            self.pc += 1
30        return self.broth
31
32    def saute(self, step, args):
33        if step == "pour": # SET
34            self.broth = self.peel(args[0])
35
36        elif step == "slice": # SPLIT
37            delim = self.peel(args[0])
38            self.ingredients = self.broth.split(delim)
39
40        elif step == "stir": # JOIN
41            delim = self.peel(args[0])
42            self.broth = delim.join(self.ingredients)
43
44        elif step == "flip": # REVERSE STR
45            self.broth = self.broth[::-1]
46
47        elif step == "toss": # REVERSE TOKENS
48            self.ingredients.reverse()
49
50        elif step == "season": # REPLACE
51            old, new = map(self.peel, args[:2])
52            self.broth = self.broth.replace(old, new)
53
54        elif step == "fillet": # SUBSTR
55            start, length = map(int, args)
56            self.broth = self.broth[start:start + length]
57
58        elif step == "flambe": # UPPER
59            self.broth = self.broth.upper()
60
61        elif step == "simmer": # LOWER
62            self.broth = self.broth.lower()
63
64        elif step == "garnish": # APPEND
65            self.broth += self.peel(args[0])
66
67        elif step == "plate": # PREPEND
68            self.broth = self.peel(args[0]) + self.broth
69
70        elif step == "taste_then": # IFSTRCONTAINS
71            substr, label = self.peel(args[0]), args[1]
72            if substr in self.broth:
73                self.pc = self.stations[label] - 1
74
75        elif step == "move_to": # GOTO
76            self.pc = self.stations[args[0]] - 1
77
78        elif step == "serve": # PRINT
79            print(self.broth)
80            self.cooking = False
81
82        else:
83            raise ValueError(f"Unknown step: {step}")
84
85    def peel(self, s): # _strip_quotes → peel
86        return s[1:-1] if s.startswith('"') and s.endswith('"') else s
```

B.2 Main experiments detail & example

In our main paper, we selected Naive and Evol-Instruct as baseline methods for comparison, as other approaches such as DataTune [9] showed limited effectiveness in our scenario (e.g., achieving only 34.95% accuracy on MedQA at the 10k scale when using irrelevant general data). Thus, we assessed how effectively these selected baselines could generate diverse and high-quality synthetic data from general-domain data.

We generated synthetic datasets at scales of 10k, 30k, and 120k, subsequently evaluating improvements in model performance across various benchmarks. Additionally, we also evaluated our initial proof-of-concept (motivation study) across the same three data scales. All benchmarks were tested under a zero-shot setting, where performance was assessed by directly parsing model-generated responses and comparing them to the ground-truth labels.

Table 5: **Comparison of synthetic data generation frameworks.** Accuracy (%) on four benchmarks across synthetic data scales (10k, 30k, 120k). Our PANGAEA framework consistently outperforms baselines, with larger improvements at increased data scales. All evaluations are performed in a zero-shot setting.

# Synthetic	Method	Benchmarks				Avg. (impr.)
		GSM8K (↑)	MedQA (↑)	FinQA (↑)	CDSL (↑)	
-	Pre-trained	5.69	28.91	6.02	0.00	10.16
	Instruction-tuned	45.03	37.31	26.68	0.57	27.40
10k	Naive	26.91	35.42	24.06	3.20	22.40 (+12.24)
	Evol-Instruct	27.36	36.29	26.68	5.22	23.89 (+13.73)
	Motivation (ours)	<u>32.14</u>	<u>36.76</u>	<u>32.08</u>	<u>8.89</u>	<u>27.47</u> (+17.31)
	PANGAEA (ours)	<u>32.52</u>	<u>37.78</u>	<u>36.44</u>	<u>11.30</u>	<u>29.51</u> (+19.35)
30k	Naive	34.72	34.24	27.46	5.51	25.48 (+15.32)
	Evol-Instruct	32.51	38.09	29.64	9.57	27.45 (+17.29)
	Motivation (ours)	<u>35.48</u>	<u>39.12</u>	<u>40.19</u>	<u>10.15</u>	<u>31.24</u> (+21.08)
	PANGAEA (ours)	<u>38.36</u>	<u>39.98</u>	<u>41.41</u>	<u>17.68</u>	<u>34.36</u> (+24.20)
120k	Naive	42.68	38.02	32.43	6.96	30.02 (+19.86)
	Evol-Instruct	38.73	42.34	33.74	13.04	31.96 (+21.80)
	Motivation (ours)	<u>46.02</u>	<u>42.58</u>	<u>43.07</u>	<u>21.31</u>	<u>38.25</u> (+28.09)
	PANGAEA (ours)	<u>48.61</u>	<u>44.62</u>	<u>50.22</u>	<u>35.36</u>	<u>44.70</u> (+34.54)

As demonstrated in Table 5, our proposed three-stage **PANGAEA** framework consistently achieves superior performance across all benchmarks and data scales compared to baseline methods, exhibiting particularly significant improvements as the data scale increases. Notably, even our initial proof-of-concept (Motivation) outperforms the Evol-Instruct baseline on most benchmarks, indicating its robustness and effectiveness in leveraging irrelevant general-domain data. However, the performance gap between Motivation and Evol-Instruct is marginal in some cases, highlighting the enhanced capability and stability introduced by our structured three-stage PANGAEA framework.

Resource & hyperparameters. For our main experiments, we utilized four NVIDIA H100 GPUs. All models were trained with Fully Sharded Data Parallel (FSDP) using the AdamW optimizer with a learning rate of 1×10^{-5} , a total batch size (batch size \times gradient accumulation steps) of 64, a sequence length of 4096, and a cosine scheduler with a warm-up ratio of 0.03 over three epochs. Inference with Deepseek-V3 was conducted using its external API, whereas inference for Llama models was performed locally.

977 **Example.** Here, we present examples of synthetic data generated using our proposed three-stage
978 PANGAEA framework. Table 6 provides examples of the general-domain data alongside the cor-
979 responding **Synth-Guide Blocks** generated for each benchmark by the Prompt Writer. Table 7
980 illustrates how these Synth-Guide Blocks are subsequently projected into **synthetic** examples.

Table 6: **Examples of Synth-Guide Blocks (τ).** Synth-Guide Blocks generated from general data, tailored specifically to each benchmark using our proposed three-stage PANGAEA framework.

General Data: Two space explorers discover two unnamed planets in an unknown galaxy and name them after themselves. Sarahn is undergoing major tectonic shifts, while Timon is geologically stable. <i>Question:</i> Which planet’s volcanic crust is hotter?	
GSM8K	<p>Scenario: Sarahn and Timon are comparing the crust temperature of their planets: Sarahn’s crust is 48 °C, Timon’s crust is half as hot (0.5 ×), and they note that the temperature difference equals twice (2 ×) Timon’s crust temperature.</p> <p>Symbols:</p> <ul style="list-style-type: none"> - A = 48 °C – Sarahn’s crust temperature - B = 0.5 – “half” factor - C = 2 – “twice” factor <p>Equations:</p> <p>(1) $D = A \times B$ (2) $T = C \times D$ (3) $E = A - D$ (4) $T = E$ Target: T = difference between the planets’ crust temperatures (°C)</p> <p>Plan: (1) Get A → (2) Find D by multiplying A and B → (3) Compute T by doubling D → (4) Calculate $T = A - D$</p>
MedQA	<p>Demographics: Two space explorers, planets named Sarahn and Timon</p> <p>Timeline: Current large geographic shift on Sarahn, stable period on Timon with no specified duration</p> <p>Key events: Continental break-apart and movement on Sarahn</p> <p>Imaging/labs: Continental break-apart and movement on Sarahn</p> <p>Pathophysiology: Geologic processes such as plate tectonics and volcanic activity, potentially leading to increased heat and volcanic crust temperature on Sarahn compared to Timon</p>
FinQA	<p>A private space exploration firm, Stellar Frontiers Inc., has discovered two previously unknown planets in a distant galaxy, naming them Sarahn and Timon after their lead explorers. Initial scans reveal stark geological differences between the two planets, presenting unique investment opportunities and risks for resource extraction ventures.</p> <ul style="list-style-type: none"> - Sarahn is undergoing significant tectonic activity, with continents fracturing and shifting rapidly. This instability suggests high volcanic activity, making the planet a prime candidate for mining rare geothermal energy resources or extracting valuable minerals formed under extreme heat and pressure. However, the volatile crust poses high operational risks, requiring advanced (and costly) drilling technology and insurance against geological disruptions. - Timon in contrast, exhibits remarkable geological stability, with no continental movement detected. This makes it an attractive site for long-term infrastructure investments, such as establishing permanent research bases or low-risk mining operations for stable mineral deposits. The lack of tectonic activity implies lower energy yields from geothermal sources but offers predictable costs and safer exploration conditions. <p>Market implications: - Sarahn: Short-term, high-reward ventures tied to volatile resources, requiring risk management strategies. - Timon: Appeals to conservative investors seeking steady returns.</p> <p>Corporate decisions: Stellar Frontiers Inc. must decide whether to license exploration rights to third parties (favoring Timon’s stability) or vertically integrate operations to exploit Sarahn’s high-risk, high-reward potential. Regulatory bodies may also intervene, imposing environmental safeguards for Sarahn’s fragile ecosystem. This scenario sets the stage for analyzing risk-adjusted returns, capital allocation strategies, and sector-specific market reactions.</p>
CDSL	<p>S – “Sarahn: hot; Timon: cold”</p> <p>Goal – Swap temperature descriptors.</p> <p>Ops – <code>split(" ; ", reverse_tokens(), join(", "))</code></p> <p>Chain – (1) split → (2) reverse tokens → (3) join</p> <p>T – “Timon: cold, Sarahn: hot”</p>

Table 7: **Example of projection output.** Domain-specific synthetic data generated by projecting Synth-Guide Blocks onto scarce domain source data using our proposed PANGAEA framework.

GSM8K	Sarahn and Timon discovered two planets, where Sarahn’s crust temperature is 48 °C, and Timon’s crust is half as hot due to lower geographic activity. If the temperature difference between their planets is twice Timon’s crust temperature, what is the difference in their crust temperatures?																														
MedQA	A 35-year-old space explorer on planet Sarahn presents with a 2-week history of worsening respiratory symptoms, including cough and shortness of breath, following a recent continental break-apart event. The patient’s oxygen saturation is 88% on room air, and a chest X-ray shows bilateral infiltrates. Which of the following is the most likely pathophysiologic mechanism contributing to the patient’s symptoms? Choose one of the following: A. Increased volcanic crust temperature leading to sulfur dioxide gas exposure B. Decreased atmospheric pressure due to planetary geographic shift C. Inhalation of particulate matter from tectonic activity D. Hypobaric hypoxia resulting from high-altitude terrain formation.																														
FinQA	<p>Context: Stellar Frontiers Inc., a pioneering private space exploration firm, has recently announced the discovery of two exoplanets, Sarahn and Timon, located in a distant galaxy. Initial geological assessments have revealed divergent characteristics that present distinct investment opportunities and operational challenges for resource extraction ventures.</p> <p>Sarahn, named after lead explorer Dr. Elena Sarahn, exhibits intense tectonic activity, with continental plates undergoing rapid fracturing and displacement. This geological volatility suggests the presence of rare minerals formed under extreme heat and pressure, such as superconductors and platinum-group metals, which are highly sought after in advanced manufacturing. However, the instability of the planet’s crust necessitates the deployment of specialized drilling equipment, which carries significant capital expenditure (CapEx) and operational risk, including potential project delays due to seismic events. These risks are further compounded by the need for comprehensive insurance coverage against geological disruptions, as outlined in Note 3 to the financial statements.</p> <p>Timon, named after geologist Marcus Timon, displays remarkable geological stability, with no detectable tectonic movement. This environment is conducive to long-term infrastructure investments, such as modular research facilities or low-risk mining operations for stable mineral deposits like iron and silicon. The predictability of Timon’s conditions reduces both CapEx and operational expenditure (OpEx), making it an attractive proposition for conservative investors, including pension funds and aerospace conglomerates seeking sustainable returns.</p> <p>The table below summarizes the projected financial implications of resource extraction ventures on Sarahn and Timon over a five-year period, adjusted for inflation and risk premiums (see Note 4 for detailed risk-adjustment methodology). All figures are presented in \$ billions, net of tax and exploration licensing fees.</p> <table><tr><td></td><td>2024</td><td>2025</td><td>2026</td><td>2027</td><td>2028</td></tr><tr><td>Sarahn Revenue</td><td>\$1.2</td><td>\$2.8</td><td>\$4.5</td><td>\$6.1</td><td>\$7.9</td></tr><tr><td>Sarahn OpEx</td><td>\$1.5</td><td>\$2.3</td><td>\$3.0</td><td>\$3.8</td><td>\$4.6</td></tr><tr><td>Timon Revenue</td><td>\$0.8</td><td>\$1.6</td><td>\$2.4</td><td>\$3.2</td><td>\$4.0</td></tr><tr><td>Timon OpEx</td><td>\$0.6</td><td>\$1.0</td><td>\$1.4</td><td>\$1.8</td><td>\$2.2</td></tr></table> <p>The financial projections underscore the divergent risk-return profiles of the two exoplanets. Sarahn’s revenue growth trajectory is steep, reflecting the high-value nature of its volatile resources, but this is offset by escalating operational costs tied to risk mitigation and technological adaptation. Timon, while offering slower revenue growth, demonstrates superior cost efficiency due to its stable environment, resulting in more predictable margins. Notably, Sarahn’s OpEx as a percentage of revenue remains persistently high, indicating the ongoing financial burden of geological risk management. In contrast, Timon’s OpEx ratio declines steadily, highlighting its suitability for long-term, low-volatility investment strategies. Regulatory considerations, particularly environmental safeguards for Sarahn’s fragile ecosystem (as discussed in Note 5), may further influence capital allocation decisions.</p> <p>Question: What is the cumulative net profit (revenue minus OpEx) for Stellar Frontiers Inc. across both exoplanets from 2024 to 2026, and how does this compare to the net profit in 2027 alone?</p>		2024	2025	2026	2027	2028	Sarahn Revenue	\$1.2	\$2.8	\$4.5	\$6.1	\$7.9	Sarahn OpEx	\$1.5	\$2.3	\$3.0	\$3.8	\$4.6	Timon Revenue	\$0.8	\$1.6	\$2.4	\$3.2	\$4.0	Timon OpEx	\$0.6	\$1.0	\$1.4	\$1.8	\$2.2
	2024	2025	2026	2027	2028																										
Sarahn Revenue	\$1.2	\$2.8	\$4.5	\$6.1	\$7.9																										
Sarahn OpEx	\$1.5	\$2.3	\$3.0	\$3.8	\$4.6																										
Timon Revenue	\$0.8	\$1.6	\$2.4	\$3.2	\$4.0																										
Timon OpEx	\$0.6	\$1.0	\$1.4	\$1.8	\$2.2																										
CDSL	Given “Saran: hot; Timon: cold”, reverse the order of the entries, swap the separators to commas, and keep the temperature descriptors unchanged. Desired Output: “Timon: cold, Saran: hot”																														

981 B.3 o1 score & influence Score

982 **o1 score.** To quantitatively assess the quality of synthetic data generated by each augmentation
 983 method, we conducted an extensive evaluation using OpenAI’s o1 model. We randomly sampled
 984 5,000 synthetic examples from each method (Naive, Evol-Instruct, and PANGAEA) and evaluated
 985 each example based on five key criteria: *difficulty*, *formatting*, *problem relevance*, *language quality*,
 986 and *clarity*. Each criterion was scored from 0 to 20 points, yielding an overall quality score between
 987 0 and 100 per sample. The detailed evaluation rubric tailored specifically for the GSM8K benchmark
 988 is presented in Fig. 24.

989 Representative examples from each augmentation method, categorized into three distinct quality
 990 levels—High (95–100), Medium (90–95), and Low (0–90)—along with their respective o1 evalua-
 991 tion scores, are shown in Tables 8, 9, and 10. These examples qualitatively illustrate the variations
 992 in synthetic data quality across methods, supporting the quantitative results summarized in Fig. 4.

```

### GSM8K Grading Rubric

**Candidate Item**
"""
{questions}
"""
---
*(Five components, equally weighted; 0-20 pts each)*

1. **Problem-Solving Potential**
- Does prompt give every fact required to compute
single, unique answer?
- Prompt must not include worked solution steps.

2. **Presentation Clarity**
- Is wording clear, unambiguous to average reader?
- Sections/tables optional; ease of understanding key.
- Prompt must *not* contain answer or derivation.

3. **Mathematical Soundness**
- Are numbers, relationships, conditions consistent
and error-free?
- Single-step or multi-step reasoning acceptable if
correct.

4. **Conciseness & Readability**
- Is wording free of unnecessary jargon/filler?
- Minor formatting quirks not penalized if comprehension
intact.

5. **Determinacy & Appropriateness**
- Does problem guarantee exactly one well-defined
answer?
- Is difficulty suitable for GSM8K (basic arithmetic,
percentages, ratios, etc.)?

**Scoring:** Sum of five component scores → total 0-100.

```

Figure 24: **Evaluation rubric for synthetic data quality (GSM8K).** This rubric, utilized by OpenAI’s o1 model, assesses each synthetic data sample across five equally-weighted criteria, with each criterion scored from 0 to 20 points, resulting in a total quality score from 0 to 100.

Table 8: **Naive Method Examples.** Representative examples of synthetic data generated by the Naive method, grouped by o1 evaluation scores (High: 95–100, Medium: 90–95, Low: 0–90).

Result	Generated Synthetic Data
Low	If Dan Deacon was 30 years old when he released his third album in 2009, how old was he when he was born in Baltimore, Maryland?
Midium	A freight train rolls 120 mi on flat prairie and then 80 mi through hills. On flat track it normally burns 2 gal of diesel per mile. The trip’s fuel gauge shows 300 gal used in total (this already covers the tiny fuel spent during brief station stops—ignore that). About how many gallons per mile does the locomotive use while on the hilly stretch?
High	At the start of a 30-day month, Li Beibei receives 60 yuan in pocket money and, spending exactly 1.5 yuan on lunch each day, how much money will she have left by the end of the month?

Table 9: **Evol-Instruct Examples.** Representative examples of synthetic data generated by the Evol-Instruct method, grouped by o1 evaluation scores (High: 95–100, Medium: 90–95, Low: 0–90).

Result	Generated Synthetic Data
Low	Ethan once sorta-borrowed \$2,500-ish from Lucas at approximately 15% interest that may or may not exist. Ethan flung \$300 back sometime while Lucas was napping. It’s now both exactly and not exactly one year later. How much random cash should Ethan hurl at Lucas so they can declare the debt mysteriously settled?
Midium	Tom is repainting two rooms in his office building: first, a rectangular conference room whose floor measures 20 ft by 30 ft and whose walls rise 12 ft high, and in that room he intends to cover only the two longer walls—each of which contains a single 4 ft × 7 ft doorway that will be left unpainted—and second, a perfectly square meeting room 15 ft on a side with walls of the same height, where he plans to coat all four walls except for three identical 3 ft × 4 ft windows (one window set into each of three different walls while the fourth wall has no openings). Ignoring ceilings, floors, trim, and any other fixtures, how many square feet of wall surface will actually receive paint?
High	Lily’s phone plan costs \$45 per month. For the first three months she pays a promotional rate of \$25 per month. In the 6th month she incurs an international roaming fee of \$18. Assuming all other months are billed at the regular rate and taxes are included in the prices, what is her total payment for the first 9 months?

Table 10: **PANGAEA Examples.** Representative examples of synthetic data generated by the PANGAEA method, grouped by o1 evaluation scores (High: 95–100, Medium: 90–95, Low: 0–90).

Result	Generated Synthetic Data
Low	Carrying the \$40 he saved over the summer, a student heads to the campus bookstore and picks out three textbooks—the first carries a price tag of \$18, the second rings up at \$4 less than that, and the third, sitting on the new-releases shelf, is exactly \$2 more than the second; with tax waived for the back-to-school promotion and a free bookmark tossed in, what is the price of the third textbook?
Midium	A runner’s GPS watch shows she covered 10 miles in 1 hour 30 minutes; given this, first convert the total time to minutes, then use that time to find her average speed in miles per hour, and finally determine (to the nearest second) her average pace per mile—how many minutes and seconds does each mile take?
High	Alex saved \$150 to cover all the costs of getting his driver’s license: the state imposes a \$50 base license fee and a \$30 road-test fee, each of which is increased by a 10% sales tax that is applied once to the respective fee, and he must also pay \$12 for the required ID photo, which is not taxed. After paying every fee and the applicable tax, how many dollars remain from Alex’s \$150 budget?

The examples provided in Tables 8, 9, and 10 clearly illustrate qualitative differences in the synthetic data generated by each method. While the Naive method often generates overly simplistic or poorly structured questions, Evol-Instruct tends to create scenarios that, although slightly improved, still frequently include irrelevant details or overly complex phrasing. In contrast, PANGEA consistently produces clear, contextually coherent, and precisely structured examples. Specifically, PANGEA-generated problems demonstrate more realistic scenarios, appropriate complexity levels, and more accurate and concise wording, resulting in higher overall evaluation scores by OpenAI’s o1 model. This qualitative distinction highlights the effectiveness of PANGEA’s structured, three-stage approach in generating synthetic data that is both domain-relevant and diverse, addressing key limitations observed in other methods.

Influence score. We measure the usefulness of the synthetic training examples generated by our PANGEA framework via influence scores estimated with the **DataInf** method of Kwon et al. [16]. We denote the domain-specific source dataset by \mathcal{D}_s and use it exclusively as the validation set when evaluating how each synthetic example affects model performance. After training our baseline model, we perform an additional round of low-rank adaptation (LoRA) fine-tuning on the union of the synthetic set \mathcal{D}_{syn} and \mathcal{D}_s , and the resulting parameters are used only to compute gradients for the influence-score estimator.

Classical influence-function approaches require the inverse of the Hessian $H(\theta)$, whose exact computation or iterative approximation is prohibitive for modern LLMs, while DataInf circumvents this bottleneck by approximating the Hessian inverse in closed form using first-order gradients and a Sherman–Morrison correction, reducing the cost to $\mathcal{O}(\sum_l n d_l)$ time and $\mathcal{O}(\max_l d_l)$ memory while retaining high correlation with exact scores.

Let the synthetic training set be $\mathcal{D} = \mathcal{D}_{\text{syn}} = \{(x_i, y_i)\}_{i=1}^n$, the validation set $\mathcal{D}^{\text{val}} = \mathcal{D}_s = \{(x_j^{\text{val}}, y_j^{\text{val}})\}_{j=1}^m$, and f_θ the LoRA-adapted model; the influence of a synthetic example (x_k, y_k) on validation performance is

$$I_{\text{DataInf}}(x_k, y_k) = \sum_{l=1}^L \frac{1}{\lambda_l} \left[\frac{1}{n} \sum_{i=1}^n \frac{(v_l^\top g_{l,i})(g_{l,i}^\top g_{l,k})}{\lambda_l + g_{l,i}^\top g_{l,i}} - v_l^\top g_{l,k} \right], \quad (4)$$

where $g_{l,i} = \nabla_{\theta_l} \ell(y_i, f_\theta(x_i))$ and $v_l = \frac{1}{m} \sum_{j=1}^m \nabla_{\theta_l} \ell(y_j^{\text{val}}, f_\theta(x_j^{\text{val}}))$, and where the damping term is $\lambda_l = 0.1 \frac{1}{n d_l} \sum_{i=1}^n g_{l,i}^\top g_{l,i}$ with d_l the parameter dimension of layer l .

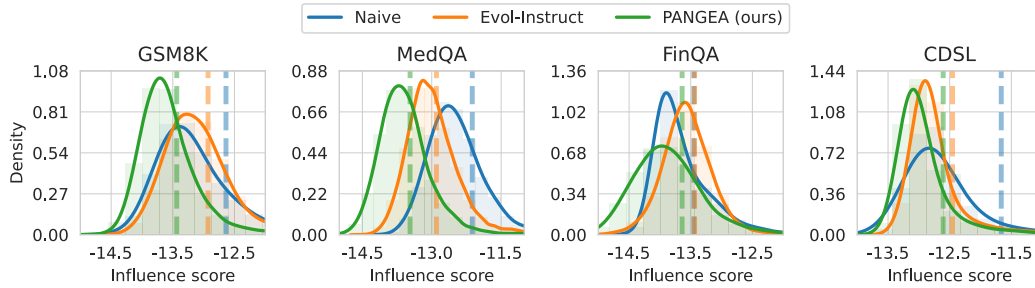


Figure 25: **Influence scores for data quality analysis.** Each histogram shows the distribution of data points for each augmentation method over influence score ranges. A kernel density estimation curve is overlaid as a solid line for better visualization, and the vertical dashed line denotes the mean.

As shown in [16], the sign of I_{DataInf} matches the direction of the validation-loss change; hence $I_{\text{DataInf}}(x_k, y_k) < 0$ implies that the sample lowers the validation loss and is therefore beneficial. Large-magnitude negative scores thus pinpoint the most influential, high-quality data.

Figure 25 plots the signed log-scaled distribution of the estimated scores, showing that PANGEA-generated synthetic examples yield larger negative influence values than all baselines and therefore lower the validation loss on \mathcal{D}_s more effectively.

1026 B.4 Embedding visualization with Cosine similarity

1027 **Embedding visualization.** To qualitatively analyze the diversity of synthetic data generated by our
 1028 framework, we visualized embeddings of 30k synthetic data points along with original source data
 1029 across all benchmarks using the `all-mpnet-base-v2` model and t-SNE projection (Fig. 26),
 1030 clearly illustrating improved diversity and coverage.

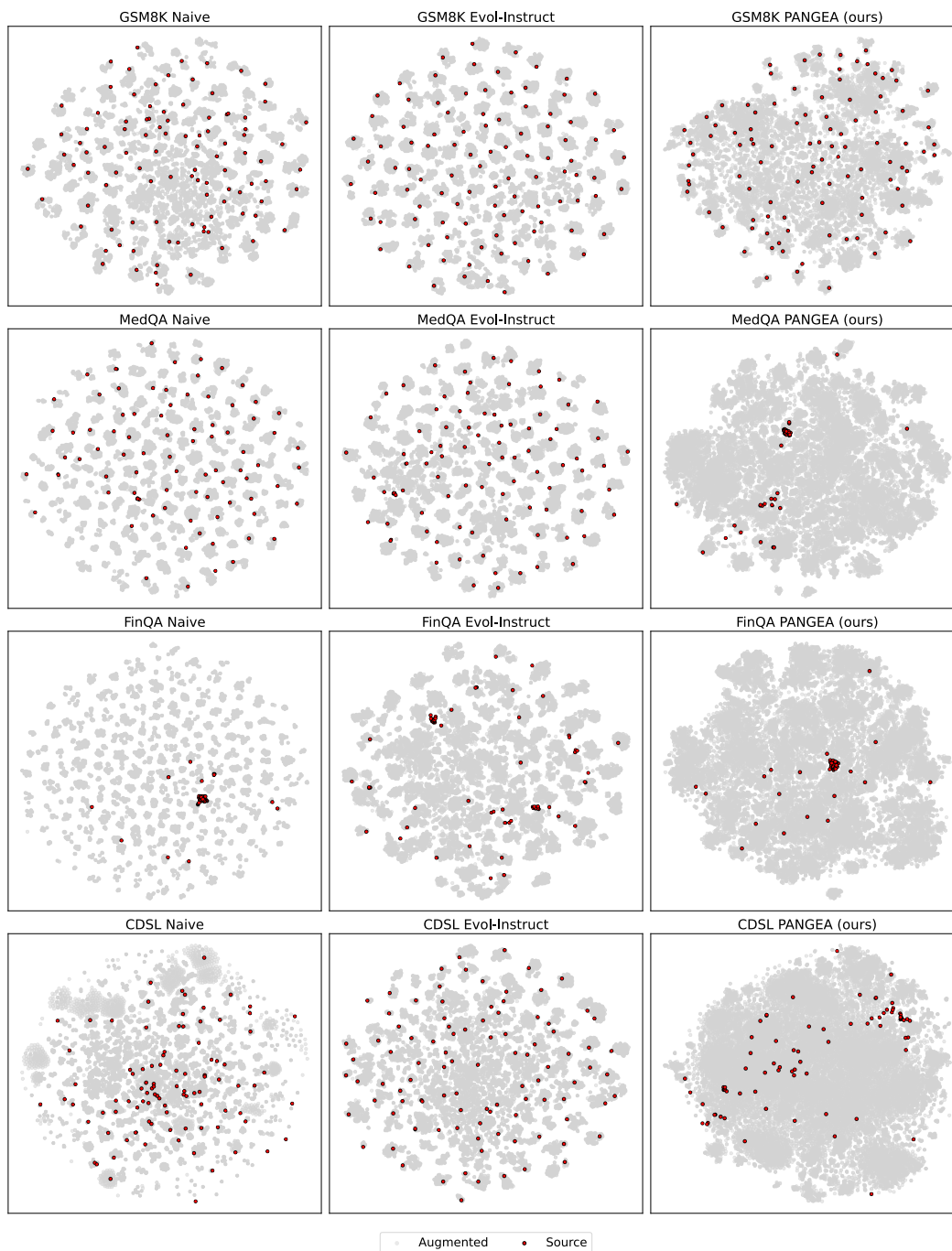


Figure 26: **Embedding Visualization of Synthetic Data Diversity.** t-SNE visualization of embeddings for 30k synthetic data points generated by Naive, Evol-Instruct, and PANGAEA methods, along with the original domain-specific source data.

Table 11: **Cosine similarity for measuring synthetic data diversity.** We compute pairwise cosine similarities between synthetic samples generated from the same source instance.

Method	GSM8K (\downarrow)	MedQA (\downarrow)	FinQA (\downarrow)	CDSL (\downarrow)
Naive	0.816 \pm 0.004	0.872 \pm 0.003	0.906 \pm 0.003	0.838 \pm 0.003
Evol-Instruct	0.776 \pm 0.003	0.847 \pm 0.002	0.742 \pm 0.002	0.767 \pm 0.008
PANGAEA	0.310\pm0.000	0.600\pm0.004	0.671\pm0.001	0.377\pm0.001

Cosine similarity. Furthermore, to quantitatively assess the diversity of synthetic data, we computed average pairwise cosine similarities between synthetic samples generated from each of the 100 source instances, with each source instance producing 100 synthetic samples. Table 11 summarizes these results, where lower cosine similarity values indicate higher intra-source diversity. PANGAEA consistently achieves substantially lower similarities across all benchmarks compared to baseline methods, clearly demonstrating its superior capability to generate numerically diverse and distinctive synthetic samples.

B.5 DataTune failed example

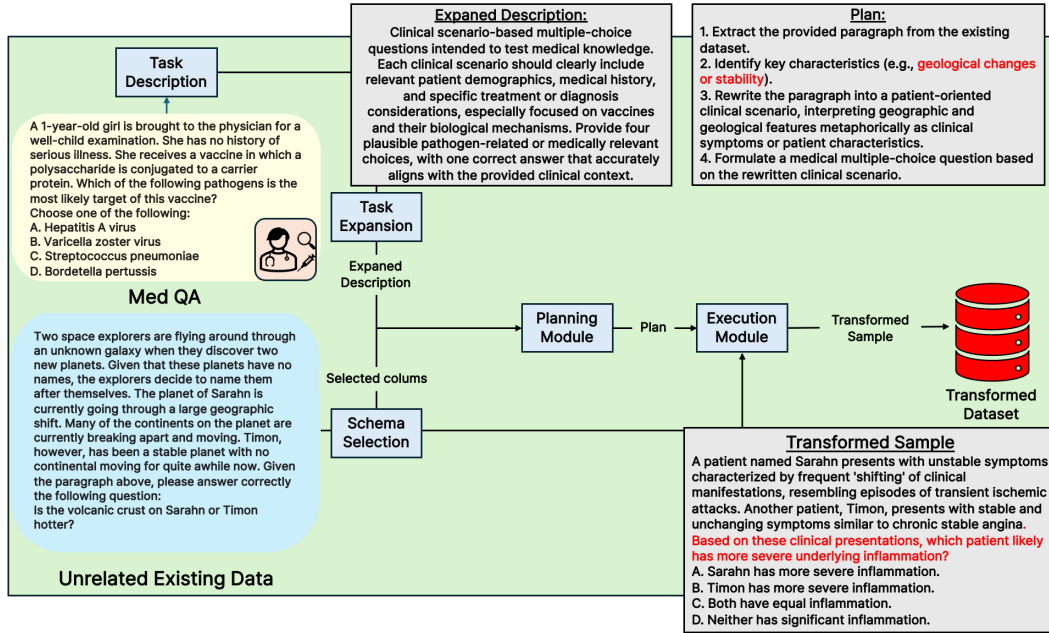


Figure 27: **Failed synthetic data examples generated by DataTune.** These examples illustrate typical issues encountered when DataTune was applied using unrelated general-domain data in our severely data-scarce setting, highlighting poor domain alignment and low-quality outcomes.

1039 To assess the performance of DataTune[9], a baseline method that augments synthetic data using re-
1040 lated existing datasets, we generated synthetic data through five module expansions as illustrated in
1041 Figure 27. While DataTune typically leverages related but possibly misaligned data, our experimen-
1042 tal setup involves an extreme scenario in which domain-specific source data (\mathcal{D}_s) is scarce, and no
1043 related existing general-domain data (\mathcal{D}_g) is available. Consequently, we had to use unrelated data
1044 for augmentation, making the application of DataTune challenging and often resulting in low-quality
1045 synthetic examples, as illustrated in Fig. 27.

1046 Empirically, when we evaluated DataTune-generated data on the MedQA benchmark at the 10k
1047 scale, the resulting model achieved an accuracy of only 34.95%, performing even worse than the
1048 Naive method, which solely relies on LLM-generated data without explicit alignment. This clearly
1049 demonstrates the limitations of employing DataTune under conditions of severe data scarcity and
1050 domain mismatch.

1051 B.6 Broader impact

1052 Our proposed synthetic data generation framework can positively impact society by effectively ad-
1053 dressing data scarcity in specialized domains, enabling improved performance and broader appli-
1054 cability of AI systems. However, we acknowledge potential negative impacts, such as the risk of
1055 generating synthetic data that unintentionally reflects biases or misinformation from general-domain
1056 sources. We therefore emphasize responsible data use, careful evaluation, and continuous monitor-
1057 ing of synthetic outputs to mitigate these risks.

1058 C Miscellaneous

1059 C.1 Related works

1060 **Synthetic Data Generation Methods.** Recent studies propose various techniques to effectively
1061 augment datasets for domain-specific tasks and LLM fine-tuning. Approaches such as Wiz-
1062 ardLM [36], WizardMath [20], and WizardCoder [21] leverage prompt-based reinforcement learning
1063 and instruction-tuning to improve LLMs’ capability to follow intricate instructions, including math-
1064 ematical reasoning, and code generation. However, these methods primarily rely on the generative
1065 capabilities of LLMs alone, often leading to the production of off-domain or lower-quality synthetic
1066 samples.

1067 TinyStories [8] demonstrated the training efficiency of smaller models using synthetic narratives
1068 generated by large-scale LLMs such as GPT-3.5 and GPT-4. Despite this effectiveness, its gener-
1069 ated data lacks structural variety, restricting scalability and limiting applicability in specialized or
1070 complex scenarios.

1071 Magpie [38] proposes an innovative method to synthesize alignment data exclusively through LLM
1072 prompting, without the need for existing aligned datasets. While advantageous for iterative and
1073 self-augmenting data improvements, this approach predominantly suits general use-cases, showing
1074 limited effectiveness in highly specialized domains. Additionally, it inherently struggles to generate
1075 data involving concepts or scenarios absent from the LLM’s original training corpus.

1076 Retrieval-augmented approaches like CRAFT [49] and DataTune [9, 27] utilize external knowledge
1077 bases or related datasets to enhance data quality. CRAFT retrieves and refines relevant text seg-
1078 ments via instruction-tuned LLMs, while DataTune [9] identifies closely related public datasets and
1079 augments data through structured modules such as Task Description, Task Expansion, Schema Se-
1080 lection, Planning, and Execution. However, both face substantial limitations when domain-relevant
1081 datasets or resources are insufficient or unavailable.

1082 ELTEX [23] addresses cybersecurity data augmentation by iteratively extracting explicit domain
1083 indicators from real-world raw data. Nonetheless, this method fundamentally relies on the presence
1084 of initial domain-specific datasets, constraining its effectiveness when such initial domain signals
1085 are absent.

1086 In task-specific settings, researchers have begun tailoring the entire data-synthesis pipeline to the
1087 unique structure of a single downstream task rather than a topical domain. For example, DISCO
1088 [50] generates phrasal perturbations with GPT-3, retaining only those pairs verified by a strong NLI
1089 teacher, significantly enhancing robustness and out-of-distribution (OOD) generalization.

1090 Similarly, CORE [51] employs a learned retriever coupled with GPT-3-driven minimal editing, ex-
1091 tracting label-flipping excerpts from a large unlabeled corpus to improve OOD performance sub-
1092 stantially (up to 4.5 pp on NLI and 6.2 pp on cross-domain sentiment tasks), modifying only a
1093 small fraction of the training data. However, these task-specific approaches inherently require either
1094 strong, domain-aligned teacher models or extensive unlabeled in-domain corpora, constraining their
1095 applicability in highly specialized or extremely data-scarce settings.

1096 C.2 Safeguards

1097 In this work, we introduced a synthetic data generation framework designed to effectively leverage
1098 general-domain data for generating diverse and high-quality domain-specific datasets. Throughout
1099 the data generation and model training processes, we ensured that our methodology does not create
1100 or utilize datasets containing sensitive, personal, or otherwise ethically problematic information.
1101 Instead, we exclusively used publicly available, openly accessible datasets, thereby circumventing
1102 privacy concerns and ethical risks.

1103 Moreover, recognizing the potential for misuse or unintended consequences associated with syn-
1104 thetic data and pretrained language models, we have implemented additional responsible-release
1105 safeguards. These include controlled data access mechanisms, clear and explicit usage guidelines,
1106 and detailed documentation aimed at ensuring that users understand both the capabilities and limita-
1107 tions of our synthetic data and associated models. Our guidelines particularly emphasize responsible
1108 and ethical use, aiming to minimize misuse risks and maximize positive societal impacts.